a second unit for executing the prefetched instruction, wherein the second unit includes an address unit for counting the address for the at least one instruction or the address for data so that the second unit skips the pseudo instruction.

34. (New) An apparatus for prefetching instructions and data of a program stored in a memory, wherein the program includes a pseudo instruction and at least one instruction, the pseudo instruction being arranged before the at least one instruction and including an address for the instruction or an address for data, the apparatus comprising:

a first unit for reading the program from the memory and detecting the pseudo instruction, wherein the first unit reads the instruction or data in accordance with the address for the instruction or the address for the data, wherein the first unit includes a buffer for storing the instruction or the data; and

a second unit for executing the stored instruction.

## REMARKS

The Office Action dated June 5, 2002 has been received and carefully noted. The preceding amendments and the following remarks are submitted as a full and complete response thereto. Claims 15 and 19-23 have been cancelled without prejudice or disclaimer. Claims 1-9, 14 and 16 have been amended. Claims 24-34 have been added. Support for the amendments can be found at page 8, lines 5-9, page 11, lines 5-14, and page 18, lines 22-23 of the present specification. Support for the new claims can be found at page 7, lines 1-3 and page 18, lines 22-33, and page 25,

lines 10-22 of the present specification. No new matter has been added or amendments. Accordingly, claims 1-14, 16-18, and 24-34 are pending in this application and are submitted for consideration.

Claims 17 and 18 were found to contain allowable subject matter. Since all rejections are addressed herein, Applicants request that claims 1-14, 16-18, and 24-34 be allowed.

Claims 1-18 were rejected under either of 35 U.S.C. § 112, first or second paragraph. In particular, it was noted that the specification of the present invention at page 7, lines 1-3, states that a pseudo instruction is defined to be handled in the same manner as a no-operation (NOP) instruction by the instruction execution unit 12. The Examiner asserted that the specification does not describe that the pseudo instruction is not executed. Furthermore, it was asserted in the Office Action that it is well known in the art that a NOP does not mean "no execution," and that it simply means when NOP is executed that no operation takes place and the NOP instruction gets executed. Therefore, claims 1, 8, 14-16, which where amended in the previous response dated May 8, 2002, to include the limitation "wherein pseudo instruction is not executed," were found to be indefinite.

Claims 1-9, 14 and 16 are amended herein, and the limitation "wherein pseudo instruction is not executed" has been deleted from the claims. Applicants submit that claims 1-14, 16-18, and 24-35 comply with the requirements of 35 U.S.C. § 112. Accordingly, Applicants request that the rejection be withdrawn.

Claims 1, 6-7, 15 and 22 were rejected under 35 U.S.C. § 102(e) as being anticipated by U.S. Patent No. 5,948,095 to Arora et al. ("Arora"). Applicants

respectfully traverse the rejection and submit that claims 1, 6-7, 15 and 22 recite subject matter not shown or described by Arora.

Claims 15 and 22 were cancelled. Claim 1, upon which claims 6-7 depend, defines a method for prefetching instructions and data of a program stored in a memory, wherein the program includes a pseudo instruction and at least one instruction. The pseudo instruction is arranged before the at least one instruction and including an address for the at least one instruction or an address for data. The method includes the steps of: reading the program from the memory; detecting the pseudo instruction with a first unit; reading the instruction or data in accordance with the address for the instruction or the address for the data with the first unit; storing the instruction or the data in a buffer; and, executing the stored instruction with a second unit.

According to the claimed configuration, a first unit (11) for processing a pseudo instruction is provided independent of a second unit (12) for executing an instruction that follows the pseudo instruction, and therefore, the risk of a mishit is diminished.

Arora is directed to system and methods for prefetching data. However, Arora fails to teach or suggest providing a first unit for processing a pseudo instruction, as defined by the claimed invention. Arora only discloses a processor for executing a prefetch instruction for prefeteching data from L0 cache (101) and storing the prefetched data into L0 cache (121). See Figure 1 of Arora.

In contrast, since the claimed invention includes the first unit (11) for processing a pseudo instruction, the second unit (12) acquires an instruction that follows the pseudo instruction without mishit, and the burden of the second unit is reduced. Aroura's processor does not have such an advantage of the claimed invention. Thus,

Applicants submit that Arora fails to show or describe each and every element of claims 1 and 6-7. Accordingly, Applicants request that the rejection be withdrawn and claims 1 and 6-7 be allowed.

Claims 2-5, 8-14, 16, 19-21 and 23 were rejected under 35 U.S.C. § 103(a) over Arora. Applicants respectfully traverse the rejection and submit that claims 2-5 and 8-14 recite subject matter not shown or described by Arora.

Claims 2-5 depend upon claim 1. Applicants submit that claims 2-5 are patentable for the same reasons as described above. Arora fails to show or suggest a first unit for processing a pseudo instruction is provided independent of a second unit for executing an instruction that follows the pseudo instruction.

Claim 8, upon which claims 9-13 depend, defines a microcontroller that includes a buffer, a first unit, and a second unit. The buffer is connected to a memory, for storing instructions and data of a program read from the memory. The program includes a pseudo instruction and at least one instruction. The pseudo instruction is arranged before the at least one instruction and includes an address for the at least one instruction or an address for data. The first unit includes a pseudo instruction detection unit, connected to the memory, for detecting the pseudo instruction included in the program read from the memory. The first unit also includes an address control unit, connected to the external memory and the pseudo instruction detection unit, for reading the instruction or data in accordance with the address for the instruction or the address for the data and storing the instruction or the data in the buffer. The second unit is connected to the buffer, for executing the instruction stored in the buffer.

Claim 14 defines a device for detecting a pseudo instruction preset before a specific instruction. The pseudo instruction includes an opcode and an operand. The device is independent of an instruction execution unit for executing the specific instruction. The device includes a detection circuit, connected to a data line, for receiving the pseudo instruction transferred on the data line and detecting the opcode included in the pseudo instruction. The device also includes a detection timing circuit, connected to the detection circuit, for calculating instruction length or the number of operands of the pseudo instruction from the opcode and determining the transfer period of the opcode based on the instruction length or the number of operands. The detection timing circuit supplies a signal for invalidating the opcode detection operation during an operand transfer period.

Claim 16 defines a microcontroller connected to a memory that stores instructions and data. The microcontroller includes an instruction execution unit and a prefetch circuit unit. The instruction execution unit is for reading instructions and data from the memory and for processing the read instructions. The prefetch circuit unit receives instructions and data read from the memory in response to a fetch signal, and detects pseudo instructions included in the instructions and data. The pseudo instruction precedes a branch instruction and indicates the existence of the branch instruction. The prefetch circuit is independent of the instruction execution unit. The prefetch circuit unit includes a prefetch buffer, a bus, a pseudo instruction detection unit, a holding circuit, a pseudo instruction buffer, and an address control unit. The prefetch buffer is connected between the instruction execution unit and the memory for temporarily storing instructions and data being transferred from the memory to the

instruction execution unit. The bus interconnects the prefetch buffer and the memory. The pseudo instruction detection unit is connected to the bus for detecting pseudo instructions among the instructions and data being transferred from the memory to the prefetch buffer. The holding circuit is connected to the bus and to the pseudo instruction detection unit, and is for storing operands of the pseudo instruction. The pseudo instruction buffer is for temporarily storing instructions and data fetched from a location in the memory which is pointed to by the branch instruction following the pseudo instruction. The address control unit is for generating the fetch signal and for generating a memory address which points to the address of a next word to be read from the memory. When the pseudo instruction detection unit detects a pseudo instruction, the instructions and data pointed to by the pseudo instruction are fetched from the memory by the address control unit and stored in the pseudo instruction buffer so that when the branch instruction following the pseudo instruction is processed by the instruction execution unit. If the branch is taken, the instructions and data pointed to by the branch instruction have been prefetched and stored in the pseudo instruction buffer.

Arora fails to show or suggest a first unit for processing a pseudo instruction that is provided independent of a second unit for executing an instruction, which follows the pseudo instruction. Thus, Applicants submit that Arora fails to show or suggest each and every element of claims 2-5 and 8-14. Accordingly, Applicants request that the rejection be withdrawn and claims 2-5 and 8-14 be allowed.

New claims 24-34 also recite a first unit for processing a pseudo instruction independent of a second unit for executing an instruction that follows the pseudo instruction.

In view of the above remarks, the Applicants respectfully submit that each of claims 1-14, 16-18, and 24-34 recites subject matter, which is neither disclosed nor suggested in the cited prior art. Applicants submit that this subject matter is more than sufficient to render the claimed invention unobvious to a person of ordinary skill in the art. Applicants therefore request that each of claims 1-14, 16-18, and 24-34 be found allowable, and this application passed to issue.

If for any reason the Examiner determines that the application is not now in condition for allowance, it is respectfully requested that the Examiner contact, by telephone, the Applicants' undersigned attorney at the indicated telephone number to arrange for an interview to expedite the disposition of this application.

In the event this paper is not timely filed, the Applicants respectfully petition for an appropriate extension of time. Any fees for such an extension together with any additional fees may be charged to Counsel's Deposit Account No. 01-2300.

Respectfully submitted,

for Brian A. Tollefson #39107
Attorney for Applicants
Registration No. 46,338

ARENT FOX KINTNER PLOTKIN & KAHN, PLLC
1050 Connecticut Avenue, N.W.,
Suite 400
Washington, D.C. 20036-5339
Tel: (202) 857-6000
Fax: (202) 638-4810

BAT:ccd

## MARKED UP COPY OF AMENDED CLAIMS

1. (Twice Amended) A method for prefetching instructions and data of [the] a program stored in a memory, wherein the program includes a pseudo instruction and at least one [of an unconditional branch instruction, a conditional branch instruction, a CALL instruction, and a data calling] instruction, the pseudo instruction being arranged before the at least one instruction and [indicating that the at least one instruction or data follows the pseudo instruction, at least one instruction address or data address being part of the pseudo instruction, wherein the pseudo instruction is not executed] including an address for the at least one instruction or an address for data, the method comprising the steps of:

reading the program from the memory;

detecting the pseudo instruction with a first unit;

[prefetching] reading the instruction or data [from the memory] in accordance with [the at least one instruction address or the data address] the address for the instruction or the address for the data with the first unit; [and]

storing the [prefetched] instruction or the data in a buffer; and

executing the stored instruction with a second unit.

2. (Amended) The method of claim 1, [further comprising a step of providing] wherein the first unit includes a pseudo instruction detection unit connected in parallel with the buffer, wherein the step of detecting the pseudo instruction includes supplying the program read from the memory to the pseudo instruction detection unit in parallel with the buffer.

3. (Amended) The method of claim 1, wherein the buffer includes first and second buffers connected in parallel with the memory, and the method further comprising a step of storing the instruction and data read from the memory in the first buffer and storing the [prefetched] instruction or data included in the detected pseudo instruction in the second buffer.

4. (Amended) The method of claim 3, [wherein the step of prefetching the instruction and data from the memory includes] further comprising the steps of:

identifying that at least one instruction following the pseudo instruction has been transferred to the first buffer with the first unit when the pseudo instruction is detected; and

prefetching the instruction or data from the memory in accordance with the at least one instruction address or data address with the first unit after the transfer of the at least one instruction to the first buffer has been identified.

5. (Amended) The method of claim [3] 4, further comprising the step of identifying that the corresponding instruction or data is stored in the second buffer in accordance with the at least one instruction address or data address when the pseudo instruction is detected, wherein the prefetch step is executed when the corresponding instruction or data is not stored in the second buffer.

6. (Amended) The method of claim 1, [wherein the step of prefetching the instruction and data from the memory includes] further comprising the steps of:

identifying that at least one instruction following the pseudo instruction has been transferred to the buffer with the first unit when the pseudo instruction is detected; and

prefetching the instruction or data from the memory in accordance with at least one instruction address or data address with the first unit after the transfer of at least one instruction to the buffer has been identified.

7. (Amended) The method of claim 6, further comprising the step of identifying that the corresponding instruction or data is stored in the buffer in accordance with the at least one instruction address or data address with the first unit when the pseudo instruction is detected, wherein the prefetch step is executed when the corresponding instruction or data is not stored in the buffer.

8. (Twice Amended) A microcontroller, comprising:

a buffer, connected to a memory, for storing instructions and data of a program [prefetched] read from the memory, wherein the program includes a pseudo instruction[, at least one of an unconditional branch instruction, a conditional branch instruction, a CALL instruction, and a data calling instruction] and at least one instruction, the pseudo instruction being arranged before the at least one instruction and [indicating that the at least one instruction or data follows the pseudo instruction, and at least one instruction address or data address being part of he pseudo instruction, and

wherein the pseudo instruction is not executed] including an address for the at least one instruction or an address for data;

[an instruction execution unit, connected to the buffer, for receiving the instruction and data from the buffer and executing a predetermined processing operation using the instruction and data;]

a first unit including,

a pseudo instruction detection unit, connected to the memory, for detecting the pseudo instruction included in the program [prefetched] read from the memory; and

an address control unit, connected to the external memory and the pseudo instruction detection unit, for [prefetching] reading the instruction or data in accordance with [at least one instruction address or data address when the pseudo instruction is detected] the address for the instruction or the address for the data and storing the instruction or the data in the buffer; and

a second unit connected to the buffer, for executing the instruction stored in the buffer.

9. (Amended) The microcontroller of claim 8, wherein the buffer includes first and second buffers connected in parallel with the memory, wherein the first buffer stores the instruction and data [prefetched] read from the memory, and the second buffer stores the instruction or data [prefetched by the address control unit] included in the detected pseudo instruction.

14. (Twice Amended) A device for detecting a pseudo instruction preset [present] before a specific instruction, wherein the pseudo instruction includes an opcode and an operand, and wherein [the pseudo instruction is not executed] the device is independent of an instruction execution unit for executing the specific instruction, the device comprising:

a detection circuit, connected to a data line, for receiving the pseudo instruction transferred on the data line and detecting the opcode included in the pseudo instruction; and

a detection timing circuit, connected to the detection circuit, for calculating instruction length or the number of operands of the pseudo instruction from the opcode and determining the transfer period of the opcode based on the instruction length or the number of operands, wherein the detection timing circuit supplies a signal for invalidating the opcode detection operation during an operand transfer period.

16. (Twice Amended) A microcontroller connected to a memory which stores instructions and data, the microcontroller comprising:

an instruction execution unit for reading instructions and data from the memory and processing the read instructions; and

a prefetch circuit unit that receives instructions and data read from the memory in response to a fetch signal, and detects pseudo instructions included in the instructions and data, wherein a pseudo instruction precedes a branch instruction and indicates the existence of the branch instruction, and wherein [the pseudo instruction is not executed] the prefetch circuit is independent of the instruction execution unit;

wherein the prefetch circuit unit includes,

a prefetch buffer connected between the instruction execution unit and the memory for temporarily storing instructions and data being transferred from the memory to the instruction execution unit,

a bus interconnecting the prefetch buffer and the memory,

a pseudo instruction detection unit connected to the bus for detecting pseudo instructions among the instructions and data being transferred from the memory to the prefetch buffer,

a holding circuit, connected to the bus and to the pseudo instruction detection unit, for storing operands of the pseudo instruction,

a pseudo instruction buffer for temporarily storing instructions and data fetched from a location in the memory which is pointed to by the branch instruction following the pseudo instruction,

an address control unit for generating the fetch signal and for generating a memory address which points to the address of a next word to be read from the memory, and wherein when the pseudo instruction detection unit detects a pseudo instruction, the instructions and data pointed to by the pseudo instruction are fetched from the memory by the address control unit and stored in the pseudo instruction buffer so that when the branch instruction following the pseudo instruction is processed by the instruction execution unit, if the branch is taken, the instructions and data pointed to by the branch instruction have been prefetched and stored in the pseudo instruction buffer.